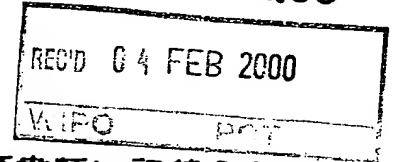


日 本 国 特 許 庁

15.12.99

JP 99/7052
EJU
PATENT OFFICE
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日

Date of Application:

1998年12月16日

出 願 番 号

Application Number:

平成10年特許願第357796号

出 願 人

Applicant (s):

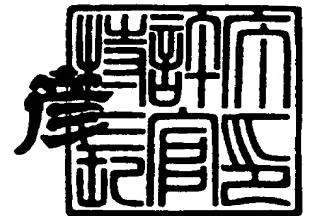
松下電器産業株式会社

**PRIORITY
DOCUMENT**
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)

2000年 1月21日

特許庁長官
Commissioner,
Patent Office

近 藤 隆 彦



出証番号 出証特平11-3095480

【書類名】 特許願

【整理番号】 2037900060

【提出日】 平成10年12月16日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 9/30350
G06F 9/32320

【発明の名称】 条件付きベクトル演算方法および条件付きベクトル演算装置

【請求項の数】 4

【発明者】

【住所又は居所】 大阪府門真市大字門真 1006 番地 松下電器産業株式会社内

【氏名】 ▲はま ▼田 真納

【発明者】

【住所又は居所】 大阪府門真市大字門真 1006 番地 松下電器産業株式会社内

【氏名】 九郎丸 俊一

【発明者】

【住所又は居所】 大阪府門真市大字門真 1006 番地 松下電器産業株式会社内

【氏名】 米澤 友紀

【発明者】

【住所又は居所】 大阪府門真市大字門真 1006 番地 松下電器産業株式会社内

【氏名】 中村 剛

【特許出願人】

【識別番号】 000005821

【氏名又は名称】 松下電器産業株式会社

【代理人】

【識別番号】 100081813

【弁理士】

【氏名又は名称】 早瀬 憲一

【電話番号】 06(380)5822

【手数料の表示】

【予納台帳番号】 013527

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9600402

【書類名】 明細書

【発明の名称】 条件付きベクトル演算方法および条件付きベクトル演算装置

【特許請求の範囲】

【請求項 1】 演算処理すべきデータの性質を判定するデータ判定工程と、前記データ判定工程の判定結果に基づき演算を行うべきか否かを判定する演算判定工程と、

前記演算判定工程の判定結果に応じ前記演算処理すべきデータに対し演算を実行して出力するかあるいは当該データを演算を実行することなく出力する演算処理工程とを含み、

条件付き演算をベクトル演算処理可能にしたことを特徴とする条件付きベクトル演算方法。

【請求項 2】 ベクトル演算命令を、ベクトル演算命令の発行によってデータの供給を開始するソースデータ供給処理と、前記ソースデータ供給処理によって供給されるデータを用いて演算を行い演算結果を出力する演算処理と、前記演算結果を順に格納する演算結果格納処理の 3 つのステージからなるパイプライン処理として実行するための制御信号を生成する制御処理工程と、

前記ソースデータ供給処理によって供給されるデータのうち第 1 のソースに供給されるデータについて逐次その状態を保持し、状態フラグとして出力する状態フラグ保持処理工程と、

前記状態フラグと前記ベクトル演算命令によって発行される条件との条件判断を行い、条件成立または不成立の情報を制御処理工程に提供する条件判定処理工程とを含み、

前記制御処理工程は、パイプライン処理の制御に加えて、前記条件判定処理の出力する情報が条件成立のときには前記演算処理を実行し、条件不成立のときには前記演算処理の演算結果として前記ソースデータ供給処理によって供給されるデータのうち第 1 のソースに供給されるデータをそのまま出力するように制御することを特徴とする条件付きベクトル演算方法。

【請求項 3】 演算処理すべきデータの性質を判定するデータ判定手段と、前記データ判定手段の判定結果に基づき演算を行うべきか否かを判定する演算

判定手段と、

前記演算判定手段の判定結果に応じ前記演算処理すべきデータに対し演算を実行して出力するかあるいは当該データを演算を実行することなく出力する演算処理手段とを含み、

条件付き演算をベクトル演算処理可能にしたことを特徴とする条件付きベクトル演算装置。

【請求項4】 ベクトル演算命令の発行によって第1ないし第Nのソースデータ（Nは2以上の整数）の供給を開始するソースデータ供給手段と、

前記ソースデータ供給手段より供給される第1ないし第Nのソースデータを一時保存する第1ないし第Nのレジスタと、

前記第1ないし第Nのレジスタの出力を用いて演算を行う演算手段と、

前記演算手段の出力する演算結果を一時保存するパイプライン・レジスタと、

前記パイプライン・レジスタの出力を順次格納する演算結果格納手段と、

前記ベクトル演算命令を前記ソースデータ供給手段によるデータの供給から前記第1ないし第Nのレジスタに格納するまでを第1ステージ、前記第1ないし第Nのレジスタの出力から前記演算手段による演算を行って前記パイプライン・レジスタに格納するまでを第2ステージ、前記パイプライン・レジスタの出力から前記演算結果格納手段に格納するまでを第3ステージとしたパイプライン処理を実行するための制御信号を生成する制御手段と、

前記第1のソースデータの性質を表わす情報を逐次保持する状態フラグ保持手段と、

前記状態フラグ保持手段の出力と前記ベクトル演算命令によって指定される条件情報より、前記ベクトル演算命令の条件が成立か不成立かを示す情報を出力する条件判定手段とを備え、

前記制御手段は、前記条件判定手段の出力する情報を受けて、条件不成立のときには前記演算手段の出力として前記第1のソースデータの値をそのまま出力し、また条件成立のときには前記演算手段による演算結果を選択して出力するようなモード選択信号を生成することを特徴とする条件付きベクトル演算装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、条件付きベクトル演算方法ならびに条件付きベクトル演算装置に関し、特に、マイクロプロセッサやデジタルシグナルプロセッサ（以下、DSP と称す）などと呼ばれるプロセッサやデータ処理装置に搭載される命令セットに用意された、ベクトル命令や条件付き演算命令を演算する演算方法、ならびにこの演算方法を用いて演算を行う演算装置の改良を図ったものに関する。

【0002】

【従来の技術】

図4は、マイクロプロセッサやDSP といった汎用的な命令セットを備えたデータ処理装置、あるいはこの種のデータ処理装置を搭載した、システムLSI と呼ばれている半導体集積回路装置を用い、演算対象のデータについて、例えば演算対象のデータがゼロであるなどの、或る条件が成立するか否かによって演算対象のデータに対して行うべき処理が異なってくるような処理を行う場合のフローチャートを示している。

この図4では、まずステップS401において演算対象のデータ402 を取得する。次にステップS403では、先のステップS401において取得したデータ402 がゼロか否かを確かめるためにゼロとデータS402との比較処理を行う。

【0003】

次のステップS404では前のステップS403の結果を受けて、その比較結果が不一致であった（No）場合にはステップS406に移り処理Bを行うが、ステップS404による判定結果が一致であった（Yes）場合には、ステップS405で演算対象のデータS402に対して処理Aを行った後、ステップS406に移る。

ここで処理Bとは、演算対象のデータ、あるいは演算対象のデータに対して処理Aの演算を施したデータを用いた処理であるとする。

このように、図4のフローチャートではステップS404の実行結果によって、以降の処理手順、即ちデータの処理の流れが異なってくる。

【0004】

ところで、一般的なマイクロプロセッサやDSP などと呼ばれるプロセッサが汎用命令セットとして備えている命令の1つに条件分岐命令がある。条件分岐命令とは、前もって実行した演算命令において演算結果の性質を示す情報をもったコンディション・コードと呼ばれるコードを生成して保持しておき、このコンディション・コードが条件分岐命令の指定する条件と一致していれば条件成立として分岐を行い、一致していなければ条件不成立として分岐を行わないという命令である。従って、図4のフローチャートにおけるステップS404のような、条件判断後の処理フローの分岐という処理を実行するには、条件分岐命令を用いるのが一般的である。

【0005】

しかしながら、図4に示したような処理を行う場合に必要となる条件分岐命令は、データ処理には直接的に何ら寄与しない動作であるにもかかわらず、数サイクルの実行サイクルを要し、さらにプログラムをパイプライン制御しているようなシステムLSI においてはパイプラインの流れが強制的に変更されてしまい、これによる処理のオーバーヘッドが増加してしまう。

【0006】

このような処理性能の低下を防ぐために、条件付き演算命令と呼ばれる命令を命令セットのなかに持つ機種がある。この条件付き演算命令は、様々な命令に分岐命令と同様な条件を付加し、指定された条件が成立したときのみその命令を実行し、不成立の場合は指定された動作を実行せずにそのまま次のステップに移るような命令である。このような命令をもつことによって、図4のステップ404のような条件分岐命令をプログラムから排除することができ、処理性能の低下を防ぐことが可能となる。

【0007】

また、この種の条件付き演算命令の改良を図ったものとして、例えば特開平8-305563号公報に示されたデータ処理装置のように、演算命令に、無条件でこれを実行するか、条件が「真」の時にこれを実行するか、条件が「偽」の時にこれを実行するか、を選択する少数のビットを付加することにより、条件付き

演算命令の命令の種類やオペランドの自由度の制限による性能低下を防止し、プログラムメモリの小容量化を可能にしたものもある。

【0008】

【発明が解決しようとする課題】

しかしながら、演算対象のデータがゼロであれば処理Aとして演算対象のデータに“1”を加算し、演算対象のデータがゼロでなければそのまま次の処理に移るような処理を、画像データのようなまとまったデータ数のデータに対して行いたい場合、従来の演算方法あるいは演算装置では、図4のフローをデータの数だけ繰り返すか、あるいは条件付き演算命令を用いてステップS404を排除したフローをデータの数だけ繰り返すか、のいずれかの方法で対処しなければならない。

【0009】

一般的にいて、上述のような条件付き演算を実行するような処理モジュールが頻繁に現われるプログラムは少ないかもしれないが、例えば、画像CODECの信号処理のような処理モジュールでは、膨大なデータに対して一定の処理ルーチンを繰り返し実行することが多く、処理ルーチンに含まれるステップ数が1ステップでも増減することによる処理性能への影響は大きい。

【0010】

本発明はこうした事情に鑑みてなされたもので、あるまとまったデータ量に対して上述したような条件付き演算を含む処理ルーチンによる処理を行う際に、処理のオーバーヘッドを削減することにより処理性能を向上することができる、条件付きベクトル演算方法および条件付きベクトル演算装置を提供することを目的とする。

【0011】

【課題を解決するための手段】

上記目的を達成するために、本願の請求項1の発明にかかる条件付きベクトル演算方法は、演算処理すべきデータの性質を判定するデータ判定工程と、前記データ判定工程の判定結果に基づき演算を行うべきか否かを判定する演算判定工程と、前記演算判定工程の判定結果に応じ前記演算処理すべきデータに対し演算を実行して出力するかあるいは当該データを演算を実行することなく出力する演算

処理工程とを含み、条件付き演算をベクトル演算処理可能にしたことを特徴とする。

【0012】

これらの処理工程によれば、条件付き演算を、演算を行うべきか否かを判定し、その判定結果に応じて演算を実行させるか否かを制御する処理と、この処理に応じて演算を行った結果もしくは演算を行わない入力データそのものを出力する演算処理とに分割することができ、これらを並行して動作でき、かつ、演算を行う場合も行わない場合もデータの流れが同一の経路を辿るので、条件付き演算を、パイプラインの処理の流れを強制的に変更することなく処理することが可能となり、これをベクトル演算にて高速に実行できる。

【0013】

また、上記目的を達成するために、本願の請求項2の発明にかかる条件付きベクトル演算方法は、ベクトル演算命令を、ベクトル演算命令の発行によってデータの供給を開始するソースデータ供給処理と、前記ソースデータ供給処理によって供給されるデータを用いて演算を行い演算結果を出力する演算処理と、前記演算結果を順に格納する演算結果格納処理の3つのステージからなるパイプライン処理として実行するための制御信号を生成する制御処理工程と、前記ソースデータ供給処理によって供給されるデータのうち第1のソースに供給されるデータについて逐次その状態を保持し、状態フラグとして出力する状態フラグ保持処理工程と、前記状態フラグと前記ベクトル演算命令によって発行される条件との条件判断を行い、条件成立または不成立の情報を制御処理工程に提供する条件判定処理工程とを含み、前記制御処理工程は、パイプライン処理の制御に加えて、前記条件判定処理の出力する情報が条件成立のときには前記演算処理を実行し、条件不成立のときには前記演算処理の演算結果として前記ソースデータ供給処理によって供給されるデータのうち第1のソースに供給されるデータをそのまま出力するように制御することを特徴とする。

【0014】

これらの処理工程によれば、ソースデータ供給処理によりベクトル命令形式で次々と供給されるデータに対して、パイプラインの第1ステージでデータの性質

(例えばゼロかゼロでないか)を状態フラグとして出力し、第2ステージでは条件判定処理工程によりその状態フラグの出力と命令の発行する条件とを用いて条件判定を行い、判定結果を演算処理の制御に用いることが可能となる。つまり条件付き演算を条件判定と条件分岐演算のパイプライン・ステージを分けることによって、条件付き演算をベクトル演算形式で処理することができる。よって、この条件付きベクトル演算を図4に示した従来の処理フローに適用することで、ステップS401およびステップS403とステップS404さらにステップS405がパイプライン的に処理されることとなり、多数のデータを連続で処理した場合には処理性能が向上する。

【0015】

また、上記目的を達成するために、本願の請求項3の発明にかかる条件付きベクトル演算装置は、演算処理すべきデータの性質を判定するデータ判定手段と、前記データ判定手段の判定結果に基づき演算を行うべきか否かを判定する演算判定手段と、前記演算判定手段の判定結果に応じ前記演算処理すべきデータに対し演算を実行して出力するかあるいは当該データを演算を実行することなく出力する演算処理手段とを含み、条件付き演算をベクトル演算処理可能にしたことを特徴とする。

【0016】

この構成によれば、条件付き演算を、演算を行うべきか否かを判定し、その判定結果に応じて演算を実行させるか否かを制御する処理と、この処理に応じて演算を行った結果もしくは演算を行わない入力データそのものを出力する演算処理とに分割することができ、これらを並行して動作でき、かつ、演算を行う場合も行わない場合もデータの流れが同一の経路を辿るので、条件付き演算を、パイプラインの流れを強制的に変更することなく処理することが可能となり、これをベクトル演算にて高速に実行できる。

【0017】

また、上記目的を達成するために、本願の請求項4の発明にかかる条件付きベクトル演算装置は、ベクトル演算命令の発行によって第1ないし第Nのソースデータ(Nは2以上の整数)の供給を開始するソースデータ供給手段と、前記ソー

スデータ供給手段より供給される第1ないし第Nのソースデータを一時保存する第1ないし第Nのレジスタと、前記第1ないし第Nのレジスタの出力を用いて演算を行う演算手段と、前記演算手段の出力する演算結果を一時保存するパイプライン・レジスタと、前記パイプライン・レジスタの出力を順次格納する演算結果格納手段と、前記ベクトル演算命令を前記ソースデータ供給手段によるデータの供給から前記第1ないし第Nのレジスタに格納するまでを第1ステージ、前記第1ないし第Nのレジスタの出力から前記演算手段による演算を行って前記パイプライン・レジスタに格納するまでを第2ステージ、前記パイプライン・レジスタの出力から前記演算結果格納手段に格納するまでを第3ステージとしたパイプライン処理を実行するための制御信号を生成する制御手段と、前記第1のソースデータの性質を表わす情報を逐次保持する状態フラグ保持手段と、前記状態フラグ保持手段の出力と前記ベクトル演算命令によって指定される条件情報より、前記ベクトル演算命令の条件が成立か不成立かを示す情報を出力する条件判定手段とを備え、前記制御手段は、前記条件判定手段の出力する情報を受けて、条件不成立のときには前記演算手段の出力として前記第1のソースデータの値をそのまま出力し、また条件成立のときには前記演算手段による演算結果を選択して出力するようなモード選択信号を生成することを特徴とする。

【0018】

この構成によれば、ソースデータ供給手段からベクトル命令形式で次々と供給されるデータに対して、パイプラインの第1ステージでデータの性質（例えばゼロかゼロでないか）を状態フラグ保持手段に格納し、第2ステージでは条件判定手段によりその状態フラグ保持手段の出力を命令の発行する条件と条件判定を行い、判定結果を演算手段の制御に用いることが可能となる。つまり条件判定と条件分岐演算のパイプライン・ステージを別のパイプライン・ステージに分けることによって、条件付き演算をベクトル演算形式で処理することができる。よって、この条件付きベクトル演算を図4に示した従来の処理フローに適用することで、ステップS401およびステップS403とステップS404さらにステップS405がパイプライン的に処理されることとなり、多数のデータを連続で処理した場合には処理性能が向上する。

【0019】

【発明の実施の形態】

以下、本発明の実施の形態について、図1ないし図3を用いて説明する。

（実施の形態1）

この実施の形態1は、演算処理すべきデータの性質を判定して、このデータの性質に応じて演算を行うか否かを制御することにより、条件付き演算をパイプライン処理、即ち、一連の処理を複数のステージに分け、ある一連の処理の第1のステージを実行し終わり第2のステージの実行中に、これと並行して次の一連の処理の第1のステージを実行する…、という処理を間断なく繰り返すベクトル演算処理、に組み込むことを可能にし、これにより、処理性能の向上を図ったものである。

【0020】

図1は本発明の実施の形態1による条件付きベクトル演算装置を示すものである。

図1において、100は本発明を適用した条件付きベクトル演算装置において、その演算を実行する演算手段としてのALU、101および102はソースデータ供給手段としてのメモリ、103および104はそれぞれメモリ101と102より供給されるソースデータを一時保存するレジスタ、105はALU100が出力する演算結果を一時的に保存するためのパイプライン・レジスタ、106はパイプライン・レジスタ105の出力を順次格納するための演算結果格納手段としてのメモリ、107はメモリ101から読み出されるソースデータの性質を表わす情報を逐次保持する状態フラグ保持手段、108は状態フラグ保持手段107が出力する状態フラグ、109は状態フラグ108と後述する条件情報115を入力とし、条件判定を行って条件の成立／不成立を示す情報114を出力する条件判定手段であり、これら、状態フラグ保持手段107および条件判定手段109により演算処理すべきデータの性質を判定するデータ判定手段120を構成している。また、110はベクトル演算命令の発行を受けて本発明の演算装置全体をベクトル命令形式でパイプライン制御するように制御信号を生成する制御手段であり、データ判定手段120の判定結果に応じて演算を行うべきか否かを判定する演算判定手段130としての機能を含み、ALU100は

この演算判定手段130 の判定結果に応じて演算処理すべきデータに対し演算を実行して出力するかあるいは演算を実行することなく出力する演算処理手段140 として動作する。111 は制御手段110 が出力するメモリ101,102,106 のメモリ制御信号、112 は制御手段110 が出力するレジスタ103,104 のライト・イネーブル信号、113 は制御手段110 が出力するALU100のモード選択信号、114 は条件判定手段109 が出力する条件成立／不成立を示す情報、115 は制御手段110 が出力する命令による条件情報である。

【0021】

ここで、本実施の形態1による条件付きベクトル演算装置の構成を用いた場合の動作を、図2にフローチャートを示した処理を行う場合を例にとって説明する。

図2では、まずステップS200においてjを“0”に初期化する。次にこのjを“0”に初期化した状態からステップS201で演算対象のデータX(j)を取得する。次にステップS202では、前のステップS201において取得したデータX(j)がゼロか否かを確認するためにゼロとデータX(j)との比較処理を行う。ステップS203では条件付き演算命令を用い、前のステップS202において決定したコンディション・フラグよりその比較結果が不一致であった（No）場合には、ステップS204によりX(j)をそのままY(j)として出力し、比較結果が一致した（Yes）場合には、ステップS205によりX(j)に“1”を加算したものをY(j)として出力し、ステップS206に移る。ステップS206は演算対象のデータである8個のデータの処理が終わるまで、ステップS201ないしステップS205の処理を繰り返すためのものであり、ステップS206においてjが“8”未満の（No）場合には、ステップS207においてjに“1”を加算し、ステップS201以降の処理を繰り返す。また、jが“8”以上の値となった（Yes）場合には、ステップS208においてY(0)ないしY(7)の値の累算を行う。

【0022】

このフローチャートは、X(j)で表わされた演算対象データについて、その値が“0”であるか否かによる条件分岐を行い、“0”以外であった場合（No）には演算結果Y(j)としてはX(j)そのものを、“0”であった場合（Yes）にはY(j)として

は $X(j)+1$ を、それぞれ採用し、以上の処理を j が“8”になるまでの間繰り返すことによって、8個の演算対象データに対し同様の条件付き演算を施す場合を示している。

【0023】

以上の処理に、本実施の形態1の構成を用いて、条件付きベクトル演算を実行したときのタイミング・チャートを図3に示した。

図3において、301は第1のソースデータとしての、メモリ101からのリードデータ、302は第2のソースデータとしての、レジスタ104に設定した固定値“1”、303はレジスタ103の出力、304は状態フラグ保持手段107の出力としての状態フラグ108、305は命令による条件情報115を示す値、306は条件判定手段109の出力としての条件成立／不成立を示す情報114、307はALU100のモード選択信号113、308はALU100の出力、309はパイプライン・レジスタ105の出力である。

【0024】

また、310はベクトル演算命令の発行によってデータの供給を開始するソースデータ供給処理としての第1ステージ、311は前記ソースデータ供給処理によって供給されるデータを用いて演算を行い演算結果を出力する演算処理としての第2ステージ、312は前記演算結果を順に格納する演算結果格納処理としての第3ステージであり、図1の制御手段110はこれら第1ステージ310、第2ステージ311、第3ステージ312をパイプライン処理として実行するように、制御信号を生成している。

【0025】

ここで図3における状態フラグ304は、第1のソースデータ301のデータの性質の情報であり、データがゼロの場合には“1”の値を、データがゼロ以外の場合には“0”の値を、それぞれ出力するものとする。また、命令による条件情報305としては、“ゼロである”という条件のコンディション・コードは“1”であるものとする。またALU100のモード選択信号が、“スルー”のときにはALU100の出力306には第1のソースデータ301を、“ADD”のときには第1のソースデータ301と第2のソースデータ302の値を加算した結果を出力するものとする。

【0026】

即ち、図3のサイクル1において、制御手段110は図1のレジスタ104に値“1”を設定するとともに、命令による条件情報305として値“1”を設定する。次に、サイクル2において、制御手段110はメモリ101よりデータX(0)として値“1”を読み出し、これをサイクル3において、レジスタ103および状態フラグ保持手段107に転送する。このサイクル3において、条件判定手段109はこの状態フラグ保持手段107に保持されている値108が“1”であり、かつ制御手段110が出力する、命令による条件情報115の値が“1”であるため、これらと比較し、条件が不成立である旨の条件成立／不成立情報114を制御手段110に出力する。これにより、制御手段110はALU100に対しスルーを選択する旨のALUのモード選択信号113を出力する。これにより、ALU100は、第1のソースデータとしてのレジスタ103の出力に対し何ら演算を行うことなくその値“1”をそのまま出力する。次に、このALU100の出力“1”は、サイクル4において、パイプライン・レジスタ105に転送され、このパイプライン・レジスタ105の出力は、メモリ106に書き込まれる。

【0027】

これらサイクル2ないし4において、データX(0)がスルー処理されるのと並行して、値“2”を有するデータX(1)がこれらより1サイクルずつずれたサイクル3ないし5において、同様に処理される。この場合も条件が不成立であるので、ALU100は、その値“2”をそのまま出力する。

【0028】

次に、サイクル4において、制御手段110はメモリ101よりデータX(2)として値“0”を読み出し、これをサイクル5において、レジスタ103および状態フラグ保持手段107に転送する。このサイクル5において、条件判定手段109はこの状態フラグ保持手段107に保持されている値108が“1”であり、かつ制御手段110が出力する、命令による条件情報115の値が“1”であるため、これらと比較し、条件が成立している旨の条件成立／不成立情報114を制御手段110に出力する。これにより、制御手段110はALU100に対し加算演算を選択する旨のALUのモード選択信号113を出力する。これにより、ALU100は、第1のソースデータと

してのレジスタ103 の出力“0”に対し“1”を加える演算を行ってその加算値“1”を出力する。次に、このALU100の出力“1”は、サイクル6において、パイプライン・レジスタ105 に転送され、このパイプライン・レジスタ105 の出力は、メモリ106 に書き込まれる。

以下、このようなパイプライン動作が値“1”，“2”，“3”，“0”，“5”を有するデータX(3)，X(4)，X(5)，X(6)，X(7)に対しても同様に行われ、パイプラインレジスタ105 の出力として、値“1”，“2”，“3”，“1”，“5”がそれぞれ得られる。

【0029】

これらの動作を処理の流れに着目して述べると、第1ステージ310 では、メモリ101 から第1のソースデータとしてのリードデータ301 を得るとともに、第2のソースデータとして、レジスタ104 に固定値“1”を設定する。

【0030】

次に、第2ステージ311 では、メモリ101 から第1のソースデータとしてのリードデータ301 をレジスタ103 に転送するとともに、その値を状態フラグ保持手段107 で“0”か否かを判定し、“0”以外であれば“0”を、“0”であれば“1”を、それぞれ状態フラグとして状態フラグ保持手段107 で保持する。そして、条件判定手段109 で状態フラグ保持手段107 で保持している状態フラグの値と制御手段110 から出力される、ベクトル演算命令による条件情報115 の値“1”とを比較し、ALU100による演算を行うべきか否かを示す条件成立／不成立情報114 を制御手段110 に出力する。これら状態フラグ保持手段107 および条件判定手段109 による処理はそれぞれ状態フラグ保持工程および条件判定処理工程に相当し、これら2つの処理はデータ判定工程に相当する。制御手段110 はこの条件成立／不成立情報114 に応じてALU100に対しALU のモード選択信号113 を出力し、条件が成立した場合にALU100がレジスタ103 の出力に対しレジスタ104 の出力である固定値“1”を加える演算を行い、条件が不成立の場合にレジスタ103 の出力になにも演算を行わず、これをスルーで出力するように制御を行い、このALU100の出力をパイプラインレジスタ105 に出力する。これらの処理において、制御手段110 の処理は演算判定工程に相当し、ALU100の処理は演算処理工程に相当

する。

そして、第3ステージ312では、ALU100の出力を一時格納したパイプラインレジスタ105の出力を、累算値を得るためのメモリ106に書き込む。

【0031】

この図3に示したタイミング・チャートにより、データに対し演算を行う場合も行わない場合もデータの流れが同じ経路を辿るので、条件付き演算を、パイプライン処理の流れを強制的に変更することなく実行できることが明らかとなり、従って、図2のフローチャートで述べた処理のうちステップS201ないしステップS207を8回繰り返す分のステップがベクトル演算で実現できることが明らかとなった。このため、条件付き演算をベクトル演算に含めることができ、これに通常のベクトル演算と同様、1命令を割り当てることができ、従って、これらサイクル1ないしサイクル11に相当するパイプライン処理を1命令で実現でき、しかもその処理を、(8サイクル) + (パイプライン遅延段数) という処理量で実現できる。

【0032】

このように、本実施の形態1によれば、処理ルーチン中に演算対象のデータの性質によって処理すべき演算内容が変わってくるような条件付き演算が含まれ、その演算対象が多数のデータにわたる場合に、条件付き演算を、条件の判定と条件分岐および演算のパイプライン・ステージを分けることにより、即ち、ALUにより条件付き演算を行う際に、演算を実行するサイクルとは別に、演算のソースとなる第1のソースデータに基づき、演算を行う条件が成立しているか否かを判定し、この判定結果に基づいて演算を実行するか否かを決定することにより、条件付き演算を、処理の流れを強制的に変更することなくベクトル演算形式で実行することが可能となり、これにより、演算対象データを連続的に処理することが可能となり、処理ステップ数の削減によるプロセッサの性能向上を図ることができる。

【0033】

なお、以上の説明では、ソースデータ供給手段としてメモリ101,102を用いたが、ソースデータ供給手段の一方をレジスタで構成した固定値としても同様の条

件付きベクトル演算装置が実現可能である。

【0034】

また、演算手段としてのALU100を累算回路で構成し、演算結果格納手段としてのメモリ106を削除し、パイプライン・レジスタ105をアキュムレート・レジスタとして構成することにより、条件付きベクトル演算の累算装置が実現可能である。

また、ソースデータの性質を表わす情報を逐次保持する状態フラグ保持手段107に保持する状態フラグとしては、ゼロかゼロ以外かを表わすゼロ・フラグの他、正負を表わすサイン・フラグなどを保持し、条件分岐の条件とすることが可能である。

さらに、膨大なデータに対して一定の処理ルーチンを繰り返し実行する処理モジュールであれば、画像データ以外のデータに対しても、その処理に使用することが可能である。

【0035】

【発明の効果】

以上のように、本願の請求項1の発明にかかる条件付きベクトル演算方法によれば、演算処理すべきデータの性質を判定するデータ判定工程と、前記データ判定工程の判定結果に基づき演算を行うべきか否かを判定する演算判定工程と、前記演算判定工程の判定結果に応じ前記演算処理すべきデータに対し演算を実行して出力するかあるいは当該データを演算を実行することなく出力する演算処理工程とを含み、条件付き演算をベクトル演算処理可能にしたので、条件付き演算を、演算を行うべきか否かを判定し、その判定結果に応じて演算を実行させるか否かを制御する処理と、この処理に応じて演算を行った結果もしくは演算を行わない入力データそのものを出力する演算処理とに分割して並行に処理することができ、かつ、その演算処理により、演算を行う場合も行わない場合もデータの流が同一の経路を辿るので、条件付き演算を、パイプラインの流れを強制的に変更することなく処理することが可能となり、これをベクトル演算にて高速に実行できる演算方法が得られる効果がある。

【0036】

また、本願の請求項2の発明にかかる条件付きベクトル演算方法によれば、ベクトル演算命令を、ベクトル演算命令の発行によってデータの供給を開始するソースデータ供給処理と、前記ソースデータ供給処理によって供給されるデータを用いて演算を行い演算結果を出力する演算処理と、前記演算結果を順に格納する演算結果格納処理の3つのステージからなるパイプライン処理として実行するための制御信号を生成する制御処理工程と、前記ソースデータ供給処理によって供給されるデータのうち第1のソースに供給されるデータについて逐次その状態を保持し、状態フラグとして出力する状態フラグ保持処理工程と、前記状態フラグと前記ベクトル演算命令によって発行される条件との条件判断を行い、条件成立または不成立の情報を制御処理工程に提供する条件判定処理工程とを含み、前記制御処理工程は、パイプライン処理の制御に加えて、前記条件判定処理の出力する情報が条件成立のときには前記演算処理を実行し、条件不成立のときには前記演算処理の演算結果として前記ソースデータ供給処理によって供給されるデータのうち第1のソースに供給されるデータをそのまま出力するよう制御するようにしたので、処理ルーチン中に演算対象のデータの性質によって処理すべき演算内容が変わってくるような条件付き演算が含まれ、その演算対象が多数のデータにわたる場合に、条件の判定と条件分岐演算のパイプライン・ステージを分けることによって、ベクトル演算形式で演算対象データを連続的に処理することが可能となり、処理ステップ数の大幅削減によるプロセッサの処理性能を向上できる演算方法を実現できる効果がある。

【0037】

また、本願の請求項3の発明にかかる条件付きベクトル演算装置によれば、演算処理すべきデータの性質を判定するデータ判定手段と、前記データ判定手段の判定結果に基づき演算を行うべきか否かを判定する演算判定手段と、前記演算判定手段の判定結果に応じ前記演算処理すべきデータに対し演算を実行して出力するかあるいは当該データを演算を実行することなく出力する演算処理手段とを備え、条件付き演算をベクトル演算処理可能にしたので、条件付き演算を、演算を行うべきか否かを判定し、その判定結果に応じて演算を実行させるか否かを制御

する処理と、この処理に応じて演算を行った結果もしくは演算を行わない入力データそのものを出力する演算処理とに分割して並行に処理することができ、かつ、その演算処理により、演算を行う場合も行わない場合もデータの流れが同一の経路を辿るので、条件付き演算を、パイプラインの流れを強制的に変更することなく処理することが可能となり、これをベクトル演算にて高速に実行できる演算装置が得られる効果がある。

【0038】

また、本願の請求項4の発明にかかる条件付きベクトル演算装置によれば、ベクトル演算命令の発行によって第1ないし第Nのソースデータ（Nは2以上の整数）の供給を開始するソースデータ供給手段と、前記ソースデータ供給手段より供給される第1ないし第Nのソースデータを一時保存する第1ないし第Nのレジスタと、前記第1ないし第Nのレジスタの出力を用いて演算を行う演算手段と、前記演算手段の出力する演算結果を一時保存するパイプライン・レジスタと、前記パイプライン・レジスタの出力を順次格納する演算結果格納手段と、前記ベクトル演算命令を前記ソースデータ供給手段によるデータの供給から前記第1ないし第Nのレジスタに格納するまでを第1ステージ、前記第1ないし第Nのレジスタの出力から前記演算手段による演算を行って前記パイプライン・レジスタに格納するまでを第2ステージ、前記パイプライン・レジスタの出力から前記演算結果格納手段に格納するまでを第3ステージとしたパイプライン処理を実行するための制御信号を生成する制御手段と、前記第1のソースデータの性質を表わす情報を逐次保持する状態フラグ保持手段と、前記状態フラグ保持手段の出力と前記ベクトル演算命令によって指定される条件情報より、前記ベクトル演算命令の条件が成立か不成立かを示す情報を出力する条件判定手段とを備え、前記制御手段は、前記条件判定手段の出力する情報を受けて、条件不成立のときには前記演算手段の出力として前記第1のソースデータの値をそのまま出力し、また条件成立のときには前記演算手段による演算結果を選択して出力するようなモード選択信号を生成するようにしたので、処理ルーチン中に演算対象のデータの性質によって処理すべき演算内容が変わってくるような条件付き演算が含まれ、その演算対象が多数のデータにわたる場合に、条件の判定と条件分岐演算のパイプライン・

ステージを分けることによって、ベクトル演算形式で演算対象データを連続的に処理することが可能となり、処理ステップ数の大幅削減によるプロセッサの処理性能を向上できる演算装置を実現できる効果がある。

【図面の簡単な説明】

【図 1】

本発明の実施の形態 1 による条件付きベクトル演算装置の構成図である。

【図 2】

本発明の実施の形態 1 による処理のフローチャートを示す図である。

【図 3】

本発明の第 1 の実施の形態にかかる図 2 のフローチャートに示した処理を本発明の第 1 の実施の形態の条件付きベクトル演算装置を用いて処理した場合のタイミングチャートを示す図である。

【図 4】

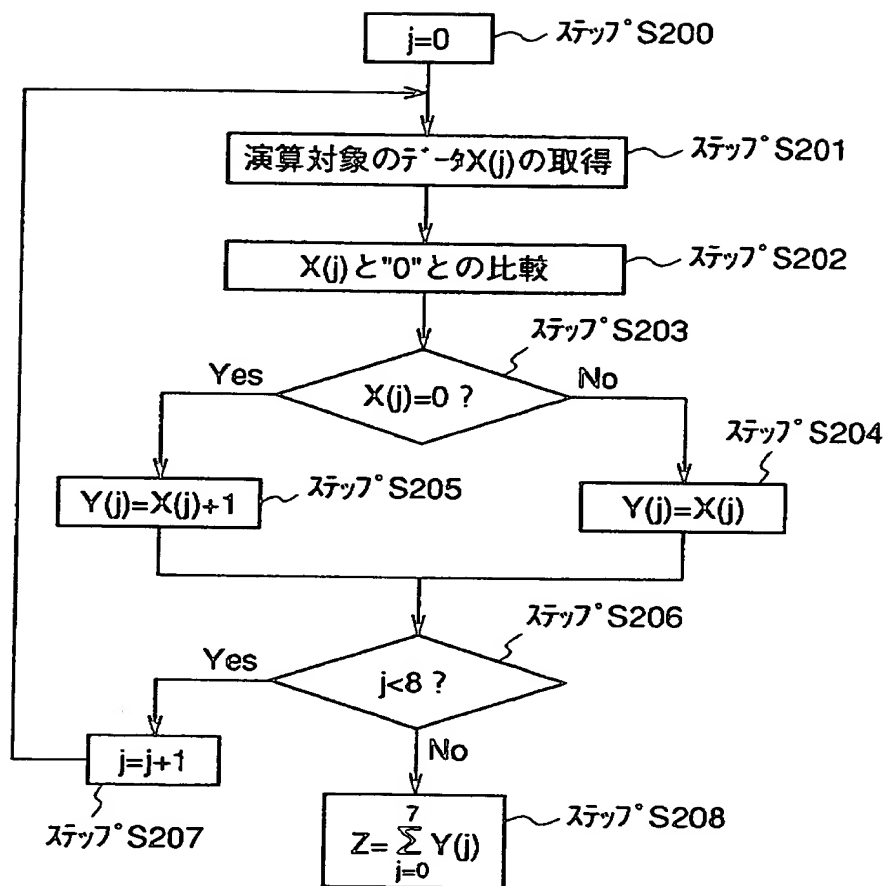
従来の条件付き演算を含む処理のフローチャートを示す図である。

【符号の説明】

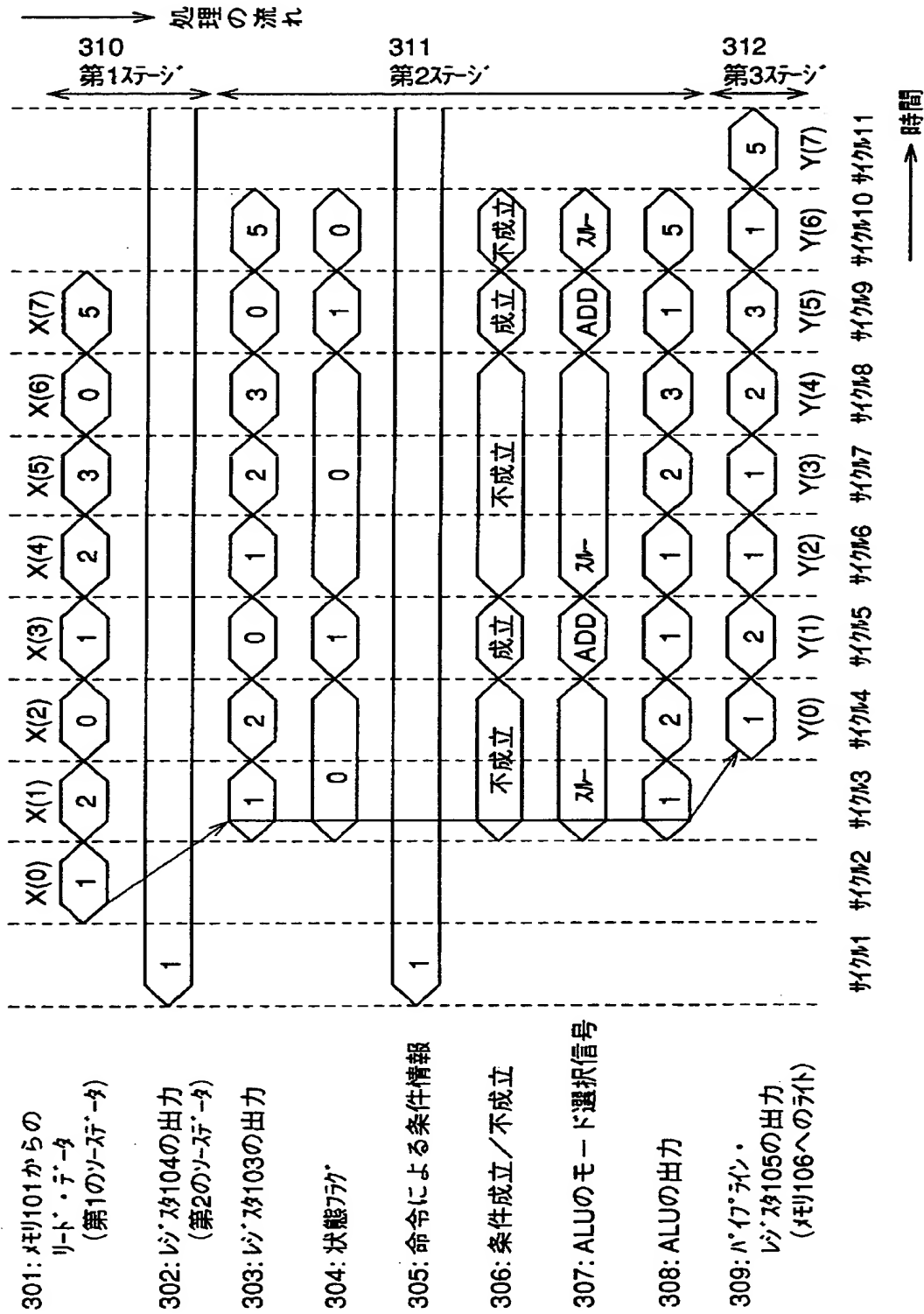
- 100 演算処理回路 (ALU)
- 101, 102 ソースデータ供給手段としてのメモリ
- 103, 104 ソースデータを一時保持するレジスタ
- 105 パイプライン・レジスタ
- 106 演算結果格納手段としてのメモリ
- 107 状態フラグ保持手段
- 108, 304 状態フラグ
- 109 条件判定手段
- 110 制御手段
- 111 メモリ制御信号
- 112 レジスタのライト・イネーブル
- 113, 307 ALU のモード選択信号
- 114, 306 条件成立／不成立情報
- 115, 305 命令による条件情報

- 120 データ判定手段
- 130 演算判定手段
- 140 演算処理手段
- 201 ~205 処理ステップ
- 301 メモリ101 からのリード・データ
- 302 レジスタ104 の出力
- 303 レジスタ103 の出力
- 308 ALU の出力
- 309 パイプライン・レジスタ105 の出力
- 310 第1ステージ
- 311 第2ステージ
- 312 第3ステージ
- 402 演算対象のデータ
- 401, 403, 404, 405, 406 処理ステップ

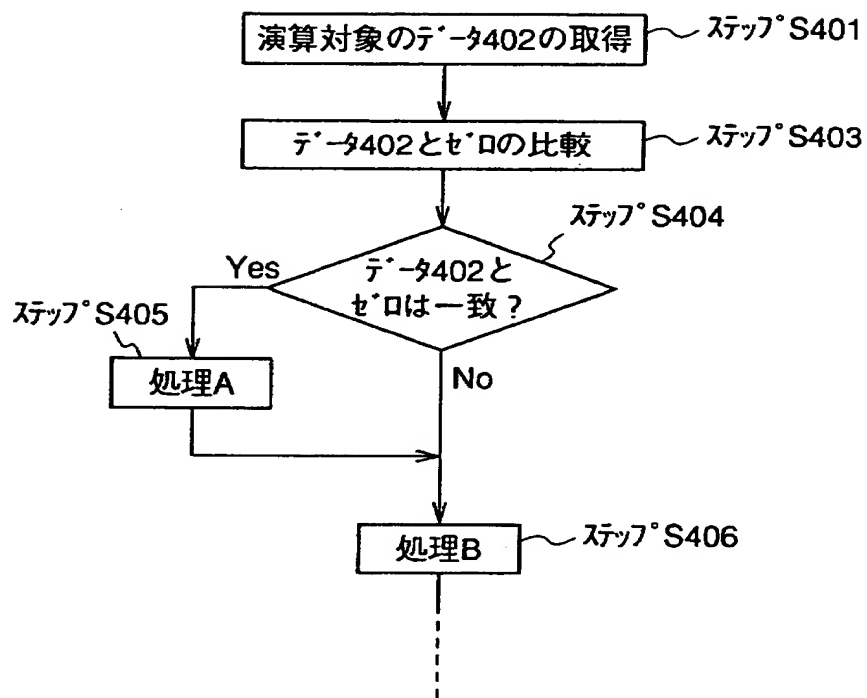
【図 2】



【図 3】



【図 4】



【書類名】 要約書

【要約】

【課題】 あるまとまったデータ量に対して条件付き演算を行う場合において、処理のオーバーヘッドを増加させることなく連続的に条件付き演算を行うことを可能にする条件付きベクトル演算方法とその装置を提供する。

【解決手段】 メモリ101 からの第1のソースデータを状態フラグ保持手段107で演算を行うべきデータであるか否かを判定して、その判定結果を状態フラグとして保持し、条件判定手段109 でその状態フラグが演算を行うべき条件が成立しているか否かを判定する。制御手段110 はこの条件成立／不成立情報に基づきALU100が演算を行うべきか否かを制御する。これにより、演算を行う場合も行わない場合もデータの流れが同じ経路を辿るので、条件付き演算をパイプライン処理の流れを乱すことなく実行でき、これをベクトル演算により高速に実行することを可能にした。

【選択図】 図1

【書類名】
【訂正書類】

職権訂正データ
特許願

<認定情報・付加情報>

【特許出願人】

【識別番号】

000005821

【住所又は居所】

大阪府門真市大字門真1006番地

【氏名又は名称】

松下電器産業株式会社

【代理人】

申請人

【識別番号】

100081813

【住所又は居所】

大阪府吹田市江の木町17番1号 江坂全日空ビル
8階 早瀬特許事務所

【氏名又は名称】

早瀬 憲一

出 願 人 履 歴 情 報

識別番号 [000005821]

1. 変更年月日	1990年 8月28日
[変更理由]	新規登録
住 所	大阪府門真市大字門真1006番地
氏 名	松下電器産業株式会社

This Page Blank (uspto)